Davide Belli
11887532
davidebelli95@gmail.com

Homework Assignment 1
Machine Learning 2, 17/18

2018-04-22

In this Homework, I discussed possible interpretations with Gabriele Cesa.

1 Problem 1

1.1 Question 1

$$\begin{aligned}
H(X,Y) &= \int\int p(x,y)(-\log p(x,y))dxdy = \\
&= \int\int p(x,y)(-\log p(x|y)p(y)))dxdy = \\
&= \int\int p(x,y)(-\log p(x|y) - \log p(y)))dxdy = \\
&= -\int\int p(x,y)\log p(x|y)dxdy - \int\int p(x,y)\log p(y)dxdy = \\
&= -\int\int p(x,y)\log p(x|y)dxdy - \int p(y)\log p(y)dy = \\
&= \mathbf{E}_{p(x,y)}[-\log p(x|y)] + \mathbf{E}_{p(y)}[-\log p(y)]dy = \\
&= H(X|Y) + H(X)
\end{aligned}$$

Conversely, by using $p(x,y) = p(y|x)p(x)$ and then the same derivations, we can find the second equality $H(X,Y) = H(X|Y) + H(X)$

1.2 Question 2

$$\begin{aligned}
I(X,Y|Z) &= H(X|Z) - H(X|Y,Z) = \\
&= \int p(z)\Big(\int\int p(x,y|z)(-\log\frac{p(x|z)p(y|z)}{p(x,y|z)}dxdy\Big)dz = \\
&= \int\int\int p(x,y,z)(-\log\frac{p(x|z)}{p(x|y,z)}dxdydz = \\
&= -\int\int\int p(x,y,z)\log p(x|z)dxdydz + \int\int\int p(x,y,z)\log p(x|y,z)dxdydz = \\
&= -\int\int p(x,z)\log p(x|z)dxdz + \int\int\int p(x,y,z)\log p(x|y,z)dxdydz = \\
&= \mathbf{E}_{p(x,z)}\Big[-\log p(x|z)\Big] - \mathbf{E}_{p(x,y,z)}\Big[-\log p(x|y,z)\Big] = \\
&= H(X|Z) - H(X|Y,Z)
\end{aligned}$$

Conversely, by using $p(x,y|z) = p(y|x,z)p(x|z)$ and then the same derivations, we can find the second equality $I(X,Y|Z) = H(Y|Z) - H(Y|X,Z)$

2 Problem 2

2.1 Question 1

$$\begin{aligned}
Mult(\boldsymbol{x},\boldsymbol{\pi}) &= \frac{M!}{x_1!\cdots x_K!}\pi_1^{x_1}\cdots\pi_K^{x_K} = \\
&= \frac{M!}{x_1!\cdots x_K!}\exp\Big(\sum_i^{K-1} x_i\log\pi_i + (M - \sum_i^{K-1} x_i)\log(1-\sum_i^{K-1}\pi_i)\Big) \\
&= \frac{M!}{x_1!\cdots x_K!}\exp\Big(\sum_i^{K-1}(x_i\log\frac{\pi_i}{1-\sum_j^{K-1}\pi_j}) + M\log(1-\sum_i^{K-1}\pi_i)\Big)
\end{aligned}$$

$$\begin{aligned}
b(x) &= \frac{M!}{x_1! \cdots x_K!} \\
T(\boldsymbol{x}) &= \boldsymbol{x} \\
\eta_i &= \log \frac{\pi_i}{1 - \sum_j^{K-1} \pi_j} \\
\pi_i &= \frac{e^{\eta_i}}{1 + \sum_j^{K-1} e^{\eta_j}} \\
A(\boldsymbol{\eta}) &= -M \log(1 - \sum_j^{K-1} \pi_j) \\
&= -M \log(1 - \frac{\sum_j^{K-1} e^{\eta_j}}{1 + \sum_j^{K-1} e^{\eta_j}}) \\
&= -M \log(\frac{1}{1 + \sum_j^{K-1} e^{\eta_j}}) \\
&= M \log(1 + \sum_j^{K-1} e^{\eta_j})
\end{aligned}$$

2.2 Question 2

$$\begin{aligned}
\mathbf{E}[x_i] &= \frac{\partial A(\boldsymbol{\eta})}{\partial \eta_i} = \\
&= \frac{M e^{\eta_i}}{1 + \sum_j^{K-1} e^{\eta_j}} = \\
&= M \frac{\frac{\pi_i}{1 - \sum_j^{K-1} \pi_j}}{1 + \frac{\sum_j^{K-1} \pi_j}{1 - \sum_j^{K-1} \pi_j}} = \\
&= M \pi_i
\end{aligned}$$

$$\begin{aligned}
Cov(x_i, x_j) &= \frac{\partial^2 A(\boldsymbol{\eta})}{\partial {\eta_i}^2} = \\
&= -M \frac{e^{\eta_i + \eta_j}}{(1 + \sum_j^{K-1} \eta_j)^2} = \\
&= -M \frac{\frac{\pi_i}{1 - \sum_j^{K-1} \pi_j} \frac{\pi_j}{1 - \sum_j^{K-1} \pi_j}}{(1 + \frac{\sum_j^{K-1} \pi_j}{1 - \sum_j^{K-1} \pi_j})^2} = \\
&= -M \pi_i \pi_j
\end{aligned}$$

2.3 Question 3

Except for a normalization constant for the Dirichlet prior:

$$\begin{aligned}
p(\boldsymbol{\pi} | \boldsymbol{\chi}, \nu) &\propto e^{\nu \boldsymbol{\chi} \boldsymbol{\eta}^T - \nu A(\boldsymbol{\eta})} = \\
&= \exp\left(\sum_i^{K-1} \chi_i \nu \log\left(\frac{\pi_i}{1 - \sum_j^{K-1} \pi_j} \right) + M\nu \log(1 - \sum_j^{K-1} \pi_j) \right) = \\
&= \prod_i^{K-1} \left(\frac{\pi_i}{1 - \sum_j^{K-1} \pi_j} \right)^{\chi_i \nu} (1 - \sum_j^{K-1} \pi_j)^{M\nu} = \\
&= \prod_i^{K-1} \left(\frac{\pi_i}{\pi_K} \right)^{\chi_i \nu} (\pi_K)^{M\nu} =
\end{aligned}$$

$$
\begin{aligned}
&= \prod_{i}^{K-1} (\pi_i)^{\chi_i \nu} (\pi_K)^{\nu(M-\sum_{j=1}^{K-1} \chi_j)} \approx \\
&\approx Dir(\boldsymbol{\pi}, \boldsymbol{\tau})
\end{aligned}
$$

$$
\begin{aligned}
&where \\
&\tau_i = \chi_i \nu + 1 \qquad\qquad i = 1, ..., K-1 \\
&\tau_K = \nu(M - \sum_{j=1}^{K-1} \chi_j) + 1
\end{aligned}
$$

2.4 Question 4

The (d) apex notation of x stands for the index of datapoints in time from 1 to D.

$$
\begin{aligned}
p(\boldsymbol{\pi}|\boldsymbol{x}, \boldsymbol{\chi}, \nu) = p(\boldsymbol{x}|\boldsymbol{\pi}) p(\boldsymbol{\pi}|\boldsymbol{\chi}, \nu) = \\
&= \frac{M!}{x_1! \cdots x_K!} \exp\Big(\sum_i^K \tau_i \log \pi_i\Big) \prod_{d=1}^{D} \exp\Big(\sum_i^K x_i^{(d)} \log \pi_i\Big) = \\
&= \frac{M!}{x_1! \cdots x_K!} \exp\Big(\sum_i^K \tau_i \log \pi_i + \sum_i^K \sum_{d=1}^{D} x_i^{(d)} \log \pi_i\Big) = \\
&= \frac{M!}{x_1! \cdots x_K!} \exp\Big(\sum_i^K (\tau_i + \sum_{d=1}^{D} x_i^{(d)}) \log \pi_i\Big) =
\end{aligned}
$$

Update rule after D datapoints:

$$
\tau_i^{(D)} = \tau_i + \sum_{d=1}^{D} x_i^{(d)}
$$

3 Problem 3

3.1 Question 1

This is an ICA model because it satisfy the following characteristic assumptions:

- sources are independent
- sources are not gaussians
- recordings are mixture of sources (plus some noise)
- there is no time delay
- datapoints are independent with respect to time

3.2 Question 2

With $t = 1, ..., T$, reminding the assumption for which datapoints at every time step are independent from datapoints at previous (and future, obviously) time steps:

$$
\begin{aligned}
p(\{s_{1,t}\}, \{s_{2,t}\}, \{x_{1,t}\}, \{x_{2,t}\}, \{x_{3,t}\}) = \prod_{t=1}^{T} & p(\{s_{1,t}\}|\nu_1) p(\{s_{2,t}\}|\nu_2) \\
& p(\{x_{\,t}\}|\{s_{1,t}\}, \{s_{2,t}\}, A_1, \sigma_1) \\
& p(\{x_{2,t}\}|\{s_{1,t}\}, \{s_{2,t}\}, A_2, \sigma_2) \\
& p(\{x_{3,t}\}|\{s_{1,t}\}, \{s_{2,t}\}, A_3, \sigma_3)
\end{aligned}
$$

3.3 Question 3

The *explaining away* phenomenom means that one of the possible causes of some data becomes less probable when a different cause for the data become more probable. For example, consider the case in which a reconstruction signal is given by the mixture of two sources. Knowing the value from one of the sources, and having the learnt model with extimated parameters for the mixture, we have a much more accurate probability distribution over the second source, as we know that the combination of the two (possibly including some noise), must result in the reconstructed signals. This is also the case of our ICA model, when the probability distribution of a source, knowing the reconstructed signals, gives also information about the probability distribution of the other source.

3.4 Question 4

a - False
b - True
c - False
d - True
e - False
f - False
g - False
h - False

3.5 Question 5

The markov blanket of a node X is set of variable nodes in the graph including its parents, its children, and the parents of its children.
The markov blanket for s_1 is the set $\{x_1, x_2, x_3\}$.
The markov blanket for x_1 is $\{s_2, s_2\}$.

3.6 Question 6

$$\begin{aligned}
p(\{x_{kt}\}|W, \{\nu_i\}) &= \prod_{t=1}^{T} p_S(\boldsymbol{W}\boldsymbol{x}_t|\{\nu_i\})\Big|det\frac{\partial \boldsymbol{S}}{\partial \boldsymbol{X}}\Big| \\
&= \prod_{t=1}^{T} p_S(\boldsymbol{W}\boldsymbol{x}_t|\{\nu_i\})|det(\boldsymbol{W})| \\
&= \prod_{t=1}^{T} p_S(\boldsymbol{s}_t|\{\nu_i\})|det(\boldsymbol{W})| \\
&= \prod_{t=1}^{T} \Big(|det(\boldsymbol{W})| \prod_{i=1}^{I} \mathcal{T}(s_{it}|0, \nu_i)\Big)
\end{aligned}$$

3.7 Question 7

$$\begin{aligned}
\log p(\{x_{kt}\}|W, \{\nu_i\}) &= \log \prod_{t=1}^{T} \Big(|det(\boldsymbol{W})| \prod_{i=1}^{I} \mathcal{T}(s_{it}|0, \nu_i)\Big) \\
&= T \log|det(\boldsymbol{W})| + \sum_{t=1}^{T}\sum_{i=1}^{I} \log \mathcal{T}(s_{it}|0, \nu_i)
\end{aligned}$$

3.8 Question 8

Stochastic Gradient Ascent (SGA), is an algorithm used to maximize the log-likelihood computed in the previous Question. To do this, in comparison with other Gradient Ascent algorithms, SGA only updates the parameters of the mixture matrix with a datapoint per iteration. An important feature in this algorithm is the usage of an activation function ϕ (often a tanh function), which describes a prior distribution over the source signals. The prior itself can be computed by integrating over this function, as we have seen in the Lab assignment. At the beginning of the algorithm itself, the mixture matrix is initialized (often randomly). In the update step of SGA, the mixture of weights is updated with a step value scaled by a learning rate parameter η. Once the algorithm converge, the product between the mixture matrix and the reconstructed signals should approximately match the original sources. Additional pre-processing often considered before performing SGA for ICA include centering and whitening of data.

3.9 Question 9

When $K >> T$ I expect overfitting. This is because too few datapoints are available to express a unique combination of features to reconstruct the sources. This also be explained with an analogy to regression problems, where we try to interpolate T datapoints with a K dimensional function. When $K >> T$, there are a lot of different functions which can perfectly interpolate those datapoints, thus we are overfitting.

4 Problem 4

4.1 Question 1

To prove this equality, we need to use d-separation on sets:

$$\begin{aligned} A &= (\boldsymbol{x}_1 ... \boldsymbol{x}_{n-1}) \\ B &= (\boldsymbol{x}_n) \\ C &= (\boldsymbol{z}_n) \end{aligned}$$

Meaning that A is d-separated from B given C. We can easily prove this by noticing that all paths from A to B pass by the node $\boldsymbol{z}_n$. If we prove that this node is blocking for the sets A and B, then d-separation is also proved. This is easily done seeing that $\boldsymbol{z}_n$ is not an end-node, it is non-collider in every path from A to B, but it also belongs to C, thus the path is blocked. Generally, this implies that every path from A to B is blocked by C.

4.2 Question 2

Similarly to the previou point, to prove this equality, we need to use d-separation on sets:

$$\begin{aligned} A &= (\boldsymbol{x}_1 ... \boldsymbol{x}_{n-1}) \\ B &= (\boldsymbol{z}_n) \\ C &= (\boldsymbol{z}_{n-1}) \end{aligned}$$

Meaning that A is d-separated from B given C. We can easily prove this by noticing that all paths from A to B pass by the node $\boldsymbol{z}_{n-1}$. If we prove that this node is blocking for the sets A and B, then d-separation is also proved. This is easily done seeing that $\boldsymbol{z}_{n-1}$ is not an end-node, it is non-collider in every path from A to B, but it also belongs to C, thus the path is blocked. Generally, this implies that every path from A to B is blocked by C.

4.3 Question 3

To prove this equality using factorization properties, it is enough to prove that $p(\boldsymbol{x}_i|\boldsymbol{z}_n, \boldsymbol{z}_{n+1}) = p(\boldsymbol{x}_i|\boldsymbol{z}_{n+1})$ for every $i = n+1...N$. From the factorization properties of bayesian networks we know that $\boldsymbol{z}_n \perp\!\!\!\perp \boldsymbol{x}_{n+1}, ..., \boldsymbol{x}_N | \boldsymbol{z}_{n+1}$, and using the Bayes' Theorem, we can derive:

$$\begin{aligned} p(\boldsymbol{x}_{n+1}, ..., \boldsymbol{x}_N|\boldsymbol{z}_n, \boldsymbol{z}_{n+1}) &= \frac{p(\boldsymbol{x}_{n+1}, ..., \boldsymbol{x}_N)p(\boldsymbol{z}_n, \boldsymbol{z}_{n+1}|\boldsymbol{x}_{n+1}, ..., \boldsymbol{x}_N)}{p(\boldsymbol{z}_n, \boldsymbol{z}_{n+1})} \\ &= \frac{p(\boldsymbol{x}_{n+1}, ..., \boldsymbol{x}_N)p(\boldsymbol{z}_n|\boldsymbol{z}_{n+1})p(\boldsymbol{z}_{n+1}|\boldsymbol{x}_{n+1}, ..., \boldsymbol{x}_N)}{p(\boldsymbol{z}_n|\boldsymbol{z}_{n+1})p(\boldsymbol{z}_{n+1})} \\ &= \frac{p(\boldsymbol{x}_{n+1}, ..., \boldsymbol{x}_N)p(\boldsymbol{z}_{n+1}|\boldsymbol{x}_{n+1}, ..., \boldsymbol{x}_N)}{p(\boldsymbol{z}_{n+1})} \\ &= p(\boldsymbol{x}_{n+1}, ..., \boldsymbol{x}_N|\boldsymbol{z}_{n+1}) \end{aligned}$$

From this, is evident that none of the variables $\boldsymbol{x}_i$ in $i = n + 1...N$ depends on $\boldsymbol{z}_n$, given $\boldsymbol{z}_{n+1}$ (looking at the graph, this is explained with every indirect dependence passing through $\boldsymbol{z}_{n+1}$ node).

4.4 Question 4

We assume that $\boldsymbol{z_{N+1}}$ is a new node in the graph which depends on $\boldsymbol{z_N}$ (since the former doesn't exist in the example graph). As we did in the previous question, using the factorization properties and Bayes' Theorem:

$$\begin{aligned} p(\boldsymbol{z}_{N+1}|\boldsymbol{z}_N, \boldsymbol{X}) &= \frac{p(\boldsymbol{z}_{N+1})p(\boldsymbol{z}_N, \boldsymbol{X}|\boldsymbol{z}_{N+1})}{p(\boldsymbol{z}_N, \boldsymbol{X})} \\ &= \frac{p(\boldsymbol{z}_{N+1})p(\boldsymbol{z}_N|\boldsymbol{X})p(\boldsymbol{X}|\boldsymbol{z}_{N+1})}{p(\boldsymbol{z}_N|\boldsymbol{X})p(\boldsymbol{X})} \\ &= \frac{p(\boldsymbol{z}_{N+1})p(\boldsymbol{X}|\boldsymbol{z}_{N+1})}{p(\boldsymbol{X})} \\ &= p(\boldsymbol{z}_{N+1}|\boldsymbol{X}) \end{aligned}$$